Stackelberg Routing on Horizontal Queueing Networks

February 8, 2012

Abstract

We study inefficiencies of capacity constrained networks due to the selfish behavior and lack of coordination of agents, by comparing social optima to Nash equilibria. Then we investigate possible strategies to reduce the inefficiency by studying the Stackelberg routing game: assuming we have control over a fraction of the flow on the network, what is a good way of routing that compliant flow so that the induced Nash equilibrium is closer to the social optimum than the initial Nash equilibrium?

Stackelberg routing on parallel link networks has been studied extensively for the class of non-decreasing latency functions, and it is shown in [?] that computing the optimal Stackelberg strategy is NP-hard in the number of links for linear latency functions.

We study Stackelberg routing for a new class of latency functions, that introduces a new model of congestion on a link. This class of latency functions arises in transportation networks for example. We show that in our setting, there are multiple Nash equilibria, and we characterize the best Nash equilibrium. Then we study the Stackelberg routing game and show that the optimal Stackelberg strategy can be computed in polynomial time for this class of latency functions on parallel link networks. The work is applied for modeling transportation networks in which a fraction of agents are compliant to routing guidance provided by a central authority, while the rest of the agents behave selfishly.

1 Introduction

Stackelberg routing on parallel link networks has been studied extensively for the class of non-decreasing latency functions, and it is known that computing the optimal Stackelberg strategy is NP-hard in the number of links. This led to considering polynomial time approximate strategies such as Largest Latency First and Scale (see [?]), and several bounds have been shown on the efficiency of these strategies.

1.1 A new class of latency functions

The class of latency functions that have been studied so far rely on very specific assumptions: if $l(x)$ is the latency on a link, where x is the flow, then l is assumed to be non-decreasing, and $x \longmapsto xl(x)$ is assumed to be convex. While this class of latency functions provides a good model for congestion for a considerable range of networks, such as communication networks, it does not accurately model horizontal queuing congestion, such as congestion on transportation networks for example. Intuitively, a given flow x on a road segment can correspond to

- either a large concentration of agents moving slowly (high density on a congested road), in which case the latency is large,
- or few cars moving fast (low density), in which case the latency is small.

this means that the latency is not uniquely determined by the flow, and depends on how congested the link is. One simple way of modeling this phenomenon is to have an additional binary argument m in the latency function $l(x, m)$ to specify whether the link is congested ($m = 1$) or is in free-flow ($m = 0$). This work considers a macroscopic model of traffic flow inspired by the model developed by Lighthill and Whitham, and Richards [?, ?], and show that the resulting latency function can be expressed in the above form $l(x, m)$. One interesting result is that the latency under congestion $l(x, 1)$ is a *decreasing* function of flow. Intuitively, as the link becomes more congested, agents slow down, so their latency increases, and the amount of flow on the link decreases.

1.2 Main results

This new class of latency functions leads to new interesting results:

- We show that for this class we do not have essential uniqueness of Nash equilibria. We also show that the set of all pure Nash equilibria can be computed in polynomial time, and we characterize in particular the best Nash equilibrium.
- We show that the optimal Stackelberg strategy can be computed in polynomial time. This result contrasts with the class of non-decreasing latency functions, for which computing the optimal Stackelberg strategy is NP-hard.

2 Related Work

3 The Model

In this section, we consider a transportation setting to motivate the choice of the class of latency functions.

3.1 Horizontal Queueing Network

We consider a network of N parallel links indexed by $n \in \{1, \ldots, N\}$, under constant positive flow demand, or rate r. Figure 1 on page 2 shows an instance of the network. The flow x_n on link n is a function of the density ρ_n, given by a triangular flux function (also referred to as fundamental diagram in transportation). It specifies the flow x_n as a function of the density on the link ρ_n by a functional relation $x_n = Q_n(\rho_n)$, where Q_n is the flux function of link n. There are various ways to model flux functions on links with horizontal queueing, most of which are concave [?]. These are usually parameterized by characteristics of the network, in particular:

- The free-flow speed on the link v_n,
- The congestion wave speed w_n,
- The maximum capacity of the link $x_n^{\max}$.

In the free flow regime (when the density on the link is less than a critical density ρ_n^{crit} that is given by $v_n \rho_n^{\text{crit}} = x_n^{\max}$) the velocity is constant and the flow increases linearly in the density $x_n = v_n \rho_n$. In the

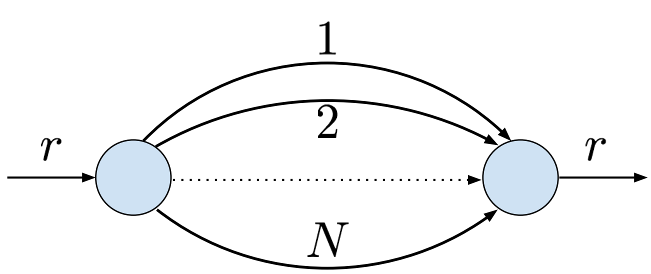


Figure 1: Network, $\mathcal{N}$, with N parallel links and demand, r.

congested regime ($\rho_n^{\text{crit}} < \rho_n \leq \rho_n^{\max}$ the density on the link is greater than the critical density and less than a maximum density given by $x_n^{\max} - w_n(\rho_n^{\max} - \rho_n^{\text{crit}}) = 0$), the flow decreases linearly in the density $x_n = x_n^{\max} - w_n(\rho_n - \rho_n^{\text{crit}}) = -w_n(\rho_n - \rho_n^{\max})$ and the velocity is decreasing [?, ?, ?].

$$x_n = \begin{cases} v_n \rho_n & \rho_n \leq \rho_n^{\text{crit}} \\ x_n^{\max} - w_n(\rho_n - \rho_n^{\text{crit}}) = -w_n(\rho_n - \rho_n^{\max}) & \rho_n^{\text{crit}} < \rho_n \leq \rho_n^{\max} \end{cases} \tag{1}$$

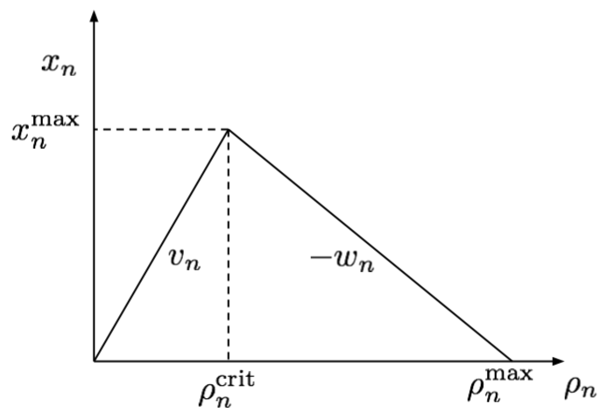


Figure 2: Fundamental diagram of traffic on link n.

We assume that the network is subject to flow demand or rate r that represents a set of non-atomic players. We denote by (N, r) a network instance with N links, rate r, and no compliant flow. We next define feasible flow assignments for network instance (N, r).

Definition 1. A flow assignment $x \in \mathbb{R}_+^N$ is feasible for instance (N, r) if $\forall n\ x_n \leq x_n^{\max}$ and $\sum_n x_n = r$

If x is a feasible flow assignment for (N, r), we denote by $\text{Supp}(x)$the support of x, that is the set of links that are used by the flow assignment

$$\text{Supp}(x) = \{n | x_n > 0\}$$

3.2 Deriving the Latency Function for a Horizontal Queueing Network

The "velocity" on link n is defined by x_n/ρ_n (flux over density), and the latency incurred by every non-atomic player on link n is given by

$$l_n(\rho_n, x_n) = \frac{L_n \rho_n}{x_n}$$

where L_n is the length of link n. In the context of traffic, velocity corresponds to physical speed of vehicles on a freeway, and latency to delay incurred while traveling through the network.

Using the expression (1) of flow as a function of the density, we can write

$$l_n(\rho_n) = \begin{cases} \frac{L_n}{v_n} & \rho_n \le \rho_n^{\text{crit}} \\ \frac{L_n}{w_n(\rho_n^{\max}/\rho_n - 1)} & \rho_n^{\text{crit}} < \rho_n \le \rho_n^{\max} \end{cases}$$

The total latency incurred by all players on a link is $x_n l_n(\rho_n) = L_n \rho_n$, and the total latency incurred by all players on the network is

$$C(\rho) = \sum_n L_n \rho_n \tag{2}$$

3.2.1 Latency as a function of flow

We can express the latency as a function of flow by introducing an integer $m_n \in \{0, 1\}$ that specifies whether link n is congested ($m_n = 1$ if n is congested and $m_n = 0$ if n is in free-flow); for this, we first extract density from flow and mode:

$$\rho_n(x_n, m_n) = \begin{cases} \frac{x_n}{v_n} & m_n = 0 \\ \frac{1}{w_n}(x_n^{\max} - x_n) + \rho_n^{\text{crit}} & m_n = 1 \end{cases}$$

this corresponds to inverting the fundamental diagram that gives the flow as a function of density. The latency is then given by the following expression:

$$l_n(x_n, m_n) = \frac{L_n \rho_n}{x_n} = \begin{cases} \frac{L_n}{v_n} & m_n = 0 \\ L_n \left(\frac{\rho_n^{\max}}{x_n} - \frac{1}{w_n} \right) & m_n = 1 \end{cases}$$

For convenience of notation, we introduce constants a_n, b_n such that

$$l_n(x_n, m_n) = \begin{cases} a_n & m_n = 0 \\ b_n \left(\frac{1}{x_n} - \frac{1}{x_n^{\max}} \right) + a_n & m_n = 1 \end{cases} \tag{3}$$

where $a_n = \frac{L_n}{v_n}$ is the free-flow latency and $b_n = L_n \rho_n^{\max}$.

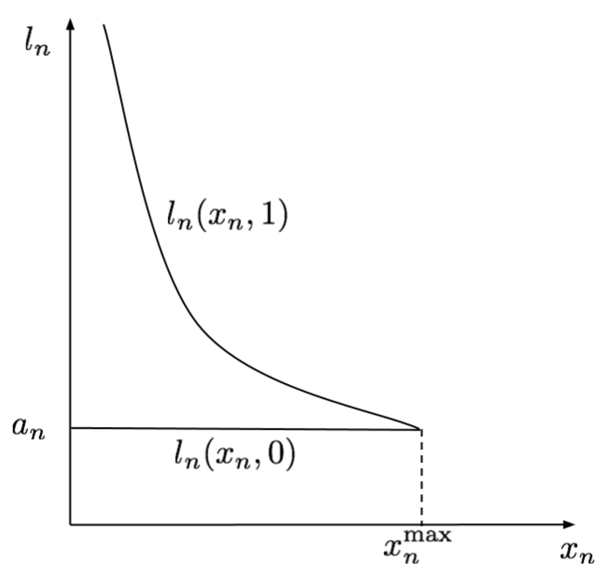


Figure 3: Latency function resulting from the a triangular flux function.

3.2.2 Properties of the latency function for a triangular flux function

Note that the latency function resulting from a concave flux function does not satisfy properties usually assumed in the Stackelberg routing literature. In particular, if we consider a triangular flux function, the latency is not an increasing function of flow: it is a constant function if the link is in free-flow, and a *decreasing* function when the link is congested. And for a given flow x_n, there are up to two possible latencies, one corresponding to the free-flow regime (low density, few agents on the link moving fast) and one to the congested regime (high density, many cars on the link moving slowly). These properties motivate the choice of the class of latency functions.

3.3 A Class of Congestion Latency Functions

We consider the class of latency functions

$$\begin{aligned} l_n : [0, x_n^{\max}] \times \{0,1\} &\longrightarrow \mathbb{R}_+ \\ (x_n, m_n) &\longmapsto l_n(x_n, m_n) \end{aligned}$$

where $x_n \in [0, x_n^{\max}]$ is the flow and $m_n \in \{0,1\}$ is the congestion state. We assume that l_n satisfies the following properties:

P1: The latency is constant in free-flow. We denote by a_n the free-flow latency where $l_n(x_n, 0) = a_n$ $\forall x_n \in [0, x_n^{\max}]$.

P2: The latency is decreasing in congestion: $x_n \longmapsto l_n(x_n, 1)$ is decreasing invertible, and $l_n(x_n^{\max}, 1) = a_n$.

Some of the known results on congestion networks do not apply to our setting: for example, the network may have multiple Nash equilibria that have different costs. In the next section we specify this result and address the issue of having multiple Nash equilibria.

4 Nash Equilibria

In this section, we characterize pure non-atomic Nash equilibria of the network (also called Wardrop equilibria in the traffic literature), which we simply refer to as Nash equilibria. We first review the essential uniqueness of Nash equilibria in the case of increasing latency functions (in the sense that all Nash equilibria have the same cost). Then we show that our class of latency functions induce multiple Nash equilibria with different costs. We show that the set of Nash equilibria can be computed in polynomial time (with respect to the number of parallel links), and we characterize the best Nash equilibrium.

4.1 Characterization of Nash Equilibria

The basic notion of Nash equilibrium on a network flow problem with a single source and single destination is given.

Definition 2. An assignment $(x, m) \in \mathbb{R}_+^N \times \{0,1\}^N$ for a parallel network instance (N, r) is at Nash equilibrium, if $\forall n$

$$x_n > 0 \Rightarrow \forall k, l_n(x_n, m_n) \leq l_k(x_k, m_k)$$

In particular, every non-atomic agent cannot improve her/his latency by switching to another link. As a consequence, all links that are in the support of x have the same latency l_0, and links that are not in the support have latency greater than l_0. Note that to fully characterize the equilibrium, one needs to specify the congestion state m, since the latency on a link depends on whether the link is congested or not. The following Lemma gives an equivalent characterization of pure Nash equilibria

Lemma 1. *If (x, m) is an assignment for a parallel network instance (N, r) at Nash equilibrium, then*

$$x_n > 0 \Rightarrow l_n(x_n, m_n) = l_0$$
$$x_n = 0 \Rightarrow l_n(0, 0) \geq l_0$$

and the total latency incurred by the network is $C(x, m) = rl_0$.

Note that links that have zero flow are necessarily in free-flow $x_n = 0 \Rightarrow m_n = 0$.

4.2 Nash Equilibria for non-decreasing latency functions

Assuming the latency functions $x_n \to l_n(x_n)$ are non-decreasing, one can show that all Nash equilibria have the same cost. Let x and x' be two assignments for (N, r) at Nash equilibrium. Let l_0, respectively l'_0 denote the common latency of all links in the support of x, respectively x'. The cost of the Nash equilibria are respectively rl_0 and rl'_0.

Assume $x \neq x'$. Then $\exists n_1, n_2$ such that

$$x_{n_1} > x'_{n_1} \geq 0 \qquad\qquad x'_{n_2} > x_{n_2} \geq 0$$

Since x is at Nash equilibrium and $n_1 \in \text{Supp}(x)$, $l_{n_1}(x_{n_1}) \leq l_{n_2}(x_{n_2})$. And since l_{n_2} is non-decreasing $l_{n_2}(x_{n_2}) \leq l_{n_2}(x'_{n_2})$. Thus $l_0 = l_{n_1}(x_{n_1}) \leq l_{n_2}(x_{n_2}) \leq l_{n_2}(x'_{n_2}) = l'_0$. Exchanging the roles of x and x' we have $l'_0 \leq l_0$. Therefore $l_0 = l'_0$ and both equilibria have the same cost.

4.3 Existence of multiple Nash equilibria for horizontal queueing networks

Let NE(N, r) denote the set of pure Nash Equilibria for network instance (N, r). To simplify the discussion, we assume without loss of generality, that the links are ordered by increasing free-flow latencies, and that free-flow latencies are different to avoid degenerate cases where the set of Nash equilibria is infinite

$$a_1 < a_2 < \ldots < a_N$$

Consider an assignment (x, m) under Nash equilibrium. Then it is clear that if link j is in the support, then links $\{1, \ldots, j-1\}$ are congested.

Lemma 2. *Let $(x, m) \in NE(N, r)$. Then $j \in Supp(x) \implies m_i = 1 \quad \forall i \in \{1, \ldots, j-1\}$*

Proof. Let $i \in \{1, \ldots, j-1\}$. Then $m_i = 0 \implies l_i = a_i < a_j \leq l_j$, which violates Lemma 1. Therefore, $m_i = 1 \quad \forall i \in \{1, \ldots, j-1\}$. □

Corollary 1. *Let $(x, m) \in NE(N, r)$. Assume that $\exists k \in Supp(x)$ such that $m_k = 0$. Then $m = (1, \ldots, \overset{k-1}{1}, \overset{k}{0}, \ldots, 0)$ and $Supp(x) = \{1, \ldots, k\}$.*

The corollary states that if some link k in the support of a Nash Equilibrium is in free flow, this completely determines the equilibrium: links $\{1, \ldots, k-1\}$ are in the support and are congested, and links $\{k+1, \ldots, N\}$ are not in the support. We will call such a Nash equilibrium (where one link in the support is in free flow) a single-link free-flow Nash equilibrium. It is clear that there are at most N such equilibria since specifying which link is in free flow completely determines the equilibrium.

There may be equilibria that are completely congested, i.e. such that $m = (1, \ldots, 1)$. We first show that there may be at most one such equilibrium.

Lemma 3. *For a given congestion state m, there is at most one assignment x such that (x, m) is at Nash equilibrium.*

Proof. In the case in which $|\text{Supp}(x)| < 2$, the result is trivial, so we assume $|\text{Supp}(x)| \geq 2$. Assume $\exists x, x', x \neq x'$ such that $(x, m), (x', m) \in \text{NE}(N, r)$. Then $\exists i, j \in \text{Supp}(x) : x_i < x'_i, x_j > x'_j$. From Corollary 1, we know that $m_i + m_j > 0$, and without loss of generality, we assume that $m_i = 1$. With this assumption, we have that $l_i > l'_i$. For the j case, we need to consider $m_j = \{0, 1\}$. If $m_j = 0$, then $l_j = l'_j$. If $m_j = 1$, then $l'_j > l_j$. Therefore, $l'_j \geq l_j$. From Lemma 1, $l'_i = l'_j$, and

$$l_i > l'_i = l'_j \geq l_j \tag{4}$$

yet from Lemma 1, we also have $l_i = l_j$, which a contradiction of (4). □

We have shown so far that there are at most $N+1$ Nash equilibria for instance (N, r) (N single-link free-flow equilibria, and one completely congested equilibrium). Next we characterize single link free flow equilibria.

4.3.1 Single link free flow Nash Equilibria

Consider a Nash equilibrium (x, m) and let $k = \max\{\text{Supp}(x)\}$ Assume $m_k = 0$ (i.e. (x, m) is a free flow Nash equilibrium). We have from Corollary 1 that links $\{1, \ldots, k-1\}$ are congested and link k is in free flow. Therefore we must have $\forall n \in \{1, \ldots, k-1\}$, $l_n(x_n, 1) = l_k(x_k, 0) = a_k$. This uniquely determines the flow on links $n \in \{1, \ldots, k-1\}$. We define this flow to be $\hat{x}_n(k)$. More precisely, for $1 \leq n < k \leq N$, $\hat{x}_n(k)$ is the unique flow that satisfies

$$l_n(\hat{x}_n(k), 1) = a_k \tag{5}$$

Note that $\hat{x}_n(k)$ is a decreasing function of k.

Now we can characterize single-link free flow equilibria. All single link free flow equilibria are of the form $(\bar{x}^{k,r}, \bar{m}^k)$ where

$$\bar{m}^k := (1, \ldots, \overset{k-1}{1}, \overset{k}{0}, \ldots, 0) \tag{6}$$

$$\bar{x}^{k,r} = (\hat{x}_1(k), \ldots, \overset{k-1}{\hat{x}_{k-1}}(k), \overset{k}{r - \sum_{n=1}^{k-1} \hat{x}_n(k)}, 0, \ldots, 0) \tag{7}$$

Lemma 4. *We have* $\text{NE}(N, r)$.

Proof. From (6) and (7), we have that $\forall i < k$, $l_i = a_j$ and $\bar{x}_i^{k,r} \in [0, x_i^{\max}] \quad \forall i \in \{1, \ldots, N\}$. By the assumption that $\bar{m}_k^k = 0$, we also have $l_k = a_k$. All links $n > k$ are not in $\text{Supp}(\bar{x}^{k,r})$, and have a latency greater than a_k. Therefore, we have that $\forall n \in \text{Supp}(\bar{x}^{k,r})$, $l_n = a_k$ and $\forall n \notin \text{Supp}(\bar{x}^{k,r})$ $l_n > a_k$, which satisfies Definition 2. □

4.3.2 Existence of a single-link free-flow Nash Equilibrium

From Lemma 4, we have an analytical representation of free flow Nash equilibria. Next, we show that if the set of Nash equilibria is non-empty, then it contains a single link free flow Nash equilibrium (i.e. a Nash equilibrium for which there is a single link in free flow).

Lemma 5. *Consider instance (N, r). If there exists a completely congested Nash equilibrium $(x, \bar{m}^{N+1}) \in \text{NE}(N, r)$, then there also exists a free flow Nash equilibrium $(\bar{x}^{j,r}, \bar{m}^j) \in \text{NE}(N, r)$ for some j.*

Proof. We first observe that the maximum demand r such that there exists a completely congested Nash equilibrium is $\sum_{n=1}^{N-1} \hat{x}_n(N) + x_N^{\max}$. Therefore, from Lemma 2, it suffices to show the following property:

$\mathbf{P_N}$: $\forall r \in [0, \sum_{n=1}^{N-1} \hat{x}_n(N) + x_N^{\max}]$, there exists a single-link free flow Nash equilibrium for instance (N, r).

We show P_N by induction on N, the size of the network. For $N = 1$, it is clear that if $0 \leq r \leq x_1^{\max}$, there is a single-link free flow Nash equilibrium $(x, m) = (r, 0)$. Now let $N \geq 1$, assume P_N is true and let us show P_{N+1}. Let $0 \leq r \leq \sum_{n=1}^{N} \hat{x}_n (N+1) + x_{N+1}^{\max}$. If $0 \leq r \leq \sum_{n=1}^{N-1} \hat{x}_n (N) + x_N^{\max}$, then by the induction hypothesis there exists a single link free flow Nash equilibrium (x, m) for instance (N, r). Then assignment (x', m') defined as $x' = (x_1, \ldots, x_N, 0)$ and $m' = (m_1, \ldots, m_N, 0)$ is clearly a single-link free flow Nash equilibrium for instance $(N+1, r)$.

Now if

$$\sum_{n=1}^{N-1} \hat{x}_n (N) + x_N^{\max} \leq r \leq \sum_{n=1}^{N} \hat{x}_n (N+1) + x_{N+1}^{\max} \tag{8}$$

, we can show that $\left(\bar{x}^{N+1,r}, \bar{m}^{N+1}\right) \in \mathrm{NE}(N+1, r)$. From Lemma 4, we only need to show that $0 \leq r - \sum_{n=1}^{N} \hat{x}_n (N+1) \leq x_{N+1}^{\max}$. We readily have the second inequality. To show the first inequality, we have

$$\begin{aligned} r &\geq \sum_{n=1}^{N-1} \hat{x}_n (N) + x_N^{\max} && \text{using (8)} \\ &\geq \sum_{n=1}^{N-1} \hat{x}_n (N+1) + x_N^{\max} && \text{since } \hat{x}_n (N) \geq \hat{x}_n (N+1) \\ &\geq \sum_{n=1}^{N-1} \hat{x}_n (N+1) + \hat{x}_N (N+1) && \text{since } x_N^{\max} \geq \hat{x}_N (N+1) \end{aligned}$$

thus we obtain the desired inequalities and $\left(\bar{x}^{N+1,r}, \bar{m}^{N+1}\right)$ is a single link free flow Nash equilibrium for instance $(N+1, r)$. This achieves the induction. □

4.4 Best Nash Equilibria

In Section 4.3, we showed that any assignment at Nash equilibrium has at most one link in free flow. We also showed that if there exists a completely congested Nash equilibrium, then there also exists a single-link free flow Nash equilibrium. We define the best Nash equilibrium to be the Nash equilibrium of least total latency

Definition 3. $\mathrm{BNE}(N, r) = \underset{(x,m) \in \mathrm{NE}(N,r)}{\arg\min} \; C(x, m)$

We now show some interesting properties of the best Nash equilibrium:

1. $\mathrm{BNE}(N, r)$ is unique.
2. $\mathrm{BNE}(N, r)$ must have a link in free flow.
3. $\mathrm{BNE}(N, r)$ has the smallest support of all Nash equilibria for demand r.

Next we show property 3. Properties 1 and 2 follow immediately.

Theorem 1. *For a parallel network instance (N, r), the best Nash equilibrium is the one that has smallest support*

$$BNE(N, r) = \underset{(x,m) \in NE(N,r)}{\arg\min} \; \{\max [Supp(x)]\}$$

Proof. To prove Theorem 1, we use two intermediate results. The first result says that a fully congested Nash equilibrium will never be the best Nash equilibrium (because there always exists a single-link free flow equilibrium, and it is better). The second result says that the better of two single-link free flow Nash equilibria will be the one with the free flow link that has smaller index.

Lemma 6. *If $(x, \bar{m}^{N+1})$ is a completely congested Nash equilibrium and $(x', \bar{m}^j)$ is a free flow Nash equilibrium, then $C\left(x', \bar{m}^j\right) \le C\left(x, \bar{m}^{N+1}\right)$.*

Lemma 7. *If $\left(\bar{x}^{r,i}, \bar{m}^i\right), \left(\bar{x}^{r,j}, \bar{m}^j\right) \in NE(N, r)$ are two free flow Nash equilibria for instance (N, r), then $i < j \Rightarrow C\left(\bar{x}^{r,i}, \bar{m}^i\right) < C\left(\bar{x}^{r,j}, \bar{m}^j\right)$*

The proofs for the lemmas above are omitted for brevity.

From Corollary 1, the complete set of candidate Nash equilibria with demand r, are the flows with one link in free flow $\left\{\bar{x}^{r,i} : i \in \{1, \dots, N\}\right\}$ and the flow with all links in congestion (x' such that $\left(x', \bar{m}^{N+1}\right) \in \mathrm{NE}(\mathscr{N}, r)$). Lemma 6 says that x' cannot be the unique minimum, and therefore a minimizing flow must exist in the set of flows with one link in free flow. Lemma 7 then says that the total latencies will decrease with decreasing free flow link index, and therefore the minimizing flow must be the Nash equilibrium flow that minimizes the index of the free flow link. □

Theorem 1 provides a simple characterization of the best Nash equilibrium for any instance (N, r). This characterization will also be useful when describing Stackelberg equilibria in Section 7.4.

4.5 Computational complexity of finding best Nash equilibria

In this section, we present a very simple algorithm for finding $\mathrm{BNE}(N, r)$, the best Nash equilibrium of a network instance (N, r) and then show the running time to be in $O\left(N^2\right)$.

Algorithm 1 Best Nash Equilibrium

```
procedure bestNE(N,r)
Inputs: Parallel network of size N; demand r
Outputs: Optimal flow/mode vectors (X,M)

for i ∈ {1,...,N}:
    let (X,M) = freeFlowConfig(N,r,i)
    if X(i) ∈ [0,c_i]:                    # c_i Equation (3)
        return (X,M)
return No-Solution


procedure freeFlowConfig(N,r,j):
Inputs: Network of size N, demand r, free flow link index j
Outputs: Flow/mode configuration (X,M)

for i ∈ {1,...,N}:
    if i < j:
        X(i) = x̂_{i,j}, M(i) = 1      # x̂_{i,j} Equation (5)
    elseif i == j:
        X(i) = r − ∑_{k<j} X(K), M(i) = 0
    else:
        X(i) = 0, M(i) = 0
return (X,M)
```

We first note from Algorithm 1 that from the definition of $\hat{x}_i(j)$, we can precompute $\hat{x}_i(j)\, \forall (i, j) \in N \times N$ in $O\left(N^2\right)$. The subroutine $freeFlowConfig$ runs in $O(N)$ time. Finally, for each loop of the $bestNE$ outer routine (with N iterations), the running time is a constant plus the running time of $freeFlowConfig$. Therefore, the overall running time of the algorithm is $O(N^2) + NO(N) = O(N^2)$.

5 Social Optima

Consider an instance (N, r) where the flow demand r does not exceed the maximum capacity of the network $r \leq \sum_n x_n^{\max}$. Since the total cost function is $C(\rho) = \sum_n L_n \rho_n$, the social optimum of the network is a solution to the following optimization problem, *Social Optimum* (SO)

$$\begin{aligned} \underset{x,m}{\text{minimize}} & \sum_n x_n l_n(x_n, m_n) \\ \text{subject to} & \sum_n x_n = r \\ & x_n \in [0, x_n^{\max}] \end{aligned}$$

The objective function is non-convex. However, we show that the solutions to this optimization problem are necessarily in free-flow $(m = 0)$, thus the social optimum can be computed by solving an equivalent linear program.

Lemma 8. *(x^*, m^*) is optimal for (SO) only if $m_i{}^* = 0 \quad \forall i \in \{1, \ldots, N\}$*

Proof. This follows immediately from the fact the latency on a link in congestion is always greater than the latency of the link in free-flow $l_n(x_n, 1) \geq l_n(x_n, 0)$ $\forall x_n \in [0, x_n^{\max}]$. □

5.1 Simple characterization of social optima

As a consequence of the previous Lemma, and using the fact that the latency is constant in free-flow $l_n(x_n, 0) = a_n$, the social optimum can be computed by solving the following equivalent linear program

$$\begin{aligned} \underset{x}{\text{minimize}} & \sum_n x_n a_n \\ \text{subject to } & \sum_n x_n = r \\ & x_n \in [0, x_n^{\max}] \end{aligned}$$

Then since the links are ordered by increasing free-flow latency $a_1 \leq \cdots \leq a_N$, the social optimum is simply given by the assignment that saturates most efficient links first. Formally, if $k_0 = \max\{k | r > \sum_{n=1}^{k} x_n^{\max}\}$ then the social optimal assignment x^* is given by

$$x^* = (x_1^{\max}, \ldots, x_{k_0}^{\max}, r - \sum_{n=1}^{k_0} x_n^{\max}, 0, \ldots, 0) \tag{9}$$

6 Price of Stability on a Two-Link Network

In this section, we characterize the loss of efficiency of Nash equilibria by comparing them to the social optimum on a simple two-link parallel network. To characterize the loss of efficiency of Nash equilibria several metrics have been used including Price of Anarchy [**?**, **?**] and Price of Stability [**?**]. The Price of Anarchy is defined as the ratio between the cost of the worst Nash equilibrium and the the social optimum cost, while the Price of Stability is defined as the ratio between the best Nash equilibrium and the social optimum cost. For the case of non-decreasing latency functions, the Price of Anarchy and the Price of Stability coincide since all Nash equilibria have the same cost by the essential uniqueness property. Since we focus our analysis on the best Nash equilibrium, we use as a metric the Price of Stability.

Consider a network instance $(2, r)$ such that $a_1 < a_2$ and $x_2^{\max} + \hat{x}_1(2) > x_1^{\max}$. Let $\text{BNE}(2, r) = (x_{\text{BNE}}(r), m_{\text{BNE}}(r))$ be the best Nash equilibrium and $(x_{\text{SO}}(r), 0)$ be the social optimum, as defined by (9). The Price of Stability is then defined as

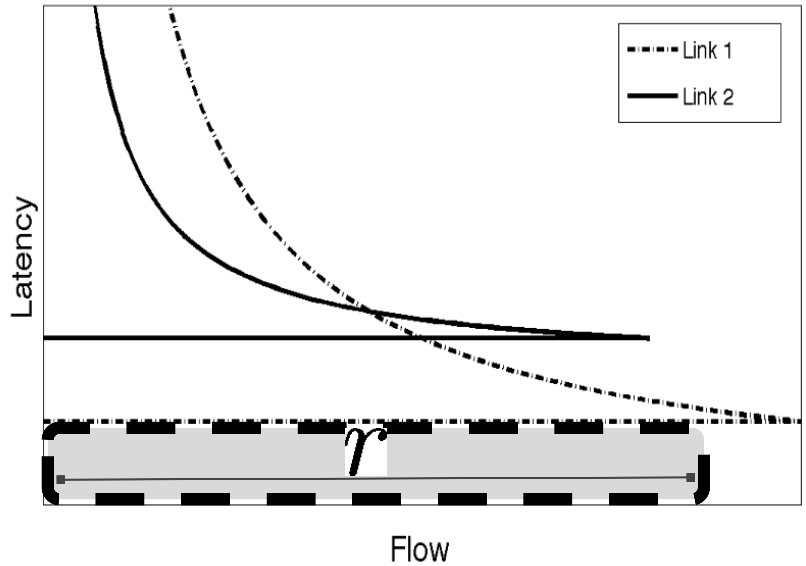


Figure 4: Flow configuration when demand is under the capacity of the fastest link.

$$\mathrm{POS}(N,r) = \frac{C\left(x_{\mathrm{BNE}}(r), m_{\mathrm{BNE}}(r)\right)}{C\left(x_{\mathrm{SO}}, 0\right)}$$

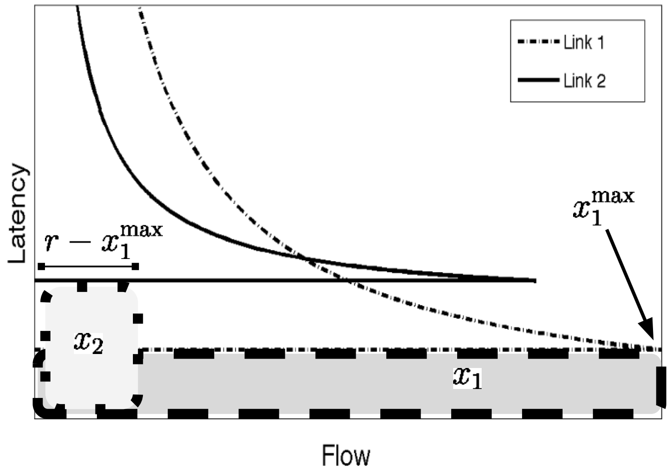
From the analysis of (9) and Algorithm 1, we only need to consider the following two cases:

Case 1: $0 \leq r \leq x_1^{\mathbf{max}}$**:** Using (9), we have $k_0 = 0$, and all the demand will be on link 1 in free flow. Similarly, from Theorem 1 we have that since link 1 can accomodate r in free flow and the support cannot be smaller than a single link, then $\mathrm{BNE}\,(2,r)$ has all flow demand on link 1 in free flow, and is equivalent to the social optimum. This is depicted in Figure 4 on page 11. In this case, the Price of Stability is equal to 1, i.e there is no drop in efficiency due to selfish routing.

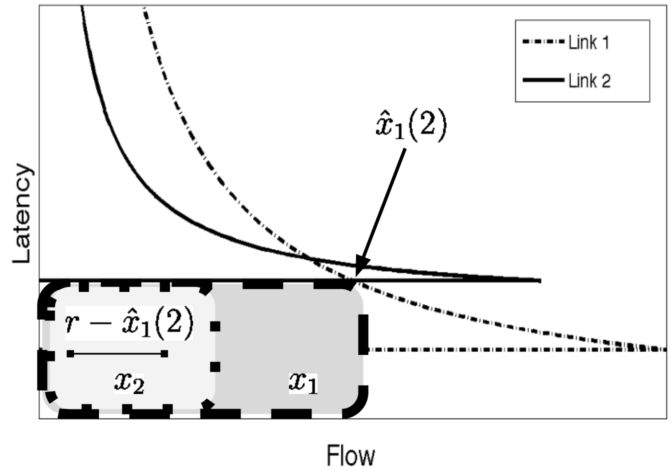
Case 2: $r > x_1^{\mathbf{max}}$**:** We know that all flow demand cannot be accomodated by link 1, and the results become more interesting. From (9), the social optimum assignment is given by $x_{SO}(r) = (x_1^{\max}, r - x_1^{\max})$. From Theorem 1 we have that $\mathrm{BNE}\,(2,r)$ has a single link in free flow. Since the total demand exceeds the capacity of link 1, then link 2 is in free flow, and link 1 is congested. Therefore $m_{NE}(r) = (1,0)$. From Algorithm 1, the corresponding flow $x_{\mathrm{NE}}(r)$ will be $\left(\hat{x}_1\left(2\right), r - \hat{x}_1\left(2\right)\right)$ (where we also tacitely assume that $x_2^{\max} \geq r - \hat{x}_1\left(2\right)$). The comparison of the social optimum and Nash equilibrium conditions are depicted in Figure 5.

Computing the Price of Stability when $r > x_1^{\max}$ reveals where the inefficiencies lie in the Nash equilibrium

$$\begin{aligned}
\mathrm{POS}(2,r) &= \frac{a_2 \hat{x}_1\left(2\right) + a_2\left(r - \hat{x}_1\left(2\right)\right)}{a_1 x_1^{\max} + a_2\left(r - x_1^{\max}\right)} \\
&= \frac{a_2 r}{r a_2 - x_1^{\max}\left(a_2 - a_1\right)} \\
&= \frac{1}{1 - \frac{x_1^{\max}}{r}\left(1 - \frac{a_1}{a_2}\right)}
\end{aligned}$$



(a) Social optimum



(b) Nash equilibrium

Figure 5: Differences in flow assignments between social optimum and Nash equilibrium when first link is beyond capacity.

First, we note that as $a_2 \to a_1$, the Price of Stability goes to 1. Intuitively, what this says is that inefficiencies in Nash equilibrium can be directly attributed to the difference in free-flow latency between the links. As the differences in free flow latency increases, so will the Price of Stability. Additionally, as the demand $r \geq x_1^{\max}$ increases, the Price of Stability decreases. This occurs because the absolute difference in total latency between social optimum and Nash equilibrium is constant for $r \geq x_1^{\max}$. Note the discontinuity in total latency for Nash equilibrium that occurs when demand exceeds the capacity of the first link ($r > x_1^{\max}$).

What we can conclude from the Price of Stability analysis of the two-link case is that selfish routing is most costly in the demand region where a free flow link is near capacity. If a controller were to anticipate a scenario where demand was slightly above this capacity, they could dramatically reduce the inefficiency of the Nash equilibrium by rerouting a small fraction of the flow.

7 Stackelberg routing

In order to reduce the inefficiency of the network, we assume that a fraction of the flow is centrally controlled, and we investigate possible strategies for improving the equilibria of the network. Leader-follower routing games have been considered in the transportation literature [?, ?] and more recently [?, ?]. The previous literature does not consider a decrease in flow on a link as a result of density buildup, while the work presented here considers the more-realistic modeling assumptions made in Section 3.

7.1 Stackelberg strategy

We consider the following problem: given a network under constant flow demand r, assume a coordinator (a central authority) has control over a fraction β of the flow: the corresponding agents are *compliant* and willing to change their routing according to the instructions they are given. The coordinator (who plays the role of the leader in the Stackelberg game) assigns the compliant flow βr according to a Stackelberg strategy s that is a feasible flow assignment for instance $(N, \beta r)$, i.e. s satisfies the followin relations:

$$s_n \leq x_n^{\max}\, \forall n \qquad \qquad \sum_n s_n = \beta r$$

We assume that the remaining flow $(1-\beta)r$ represents selfish players (who play the role of followers in the Stackelberg game), who will choose their routes after the Stackelberg strategy s is revealed. This induces an assignment $(t(s), m(s))$ of the selfish flow at Nash equilibrium, and we assume that the assignment s of compliant agents *is not affected* after introducing the non-compliant flow on the network. Since s may induce multiple Nash equilibria, we define the assignment $(t(s), m(s))$ to be the best such equilibrium.

To characterize this Nash equilibrium, which we refer to as the *induced equilibrium* by strategy s, we note that the flow on link n is simply $s_n + t_n(s)$, and we have for all n:

$$t_n(s) > 0 \Rightarrow \forall k, l_n\left(s_n + t_n(s), m_n(s)\right) \leq l_k\left(s_k + t_k(s), m_k(s)\right)$$

and bBy Lemma 1, all links that are in the support of the selfish flow assignment $t(s)$ have a common latency l_0 in the induced equilibrium, and links that are not in the support have latency greater than or equal to l_0.

This can be summarized in the following definition of a Stackelberg strategy for instance (N, r, β) of a network with N links under flow demand r, and fraction of compliance β.

Definition 4. A Stackelberg strategy for instance (N, r, β) is an assignment s of the compliant flow βr that is feasible for the instance $(N, \beta r)$, and which induces best Nash equilibrium $(t(s), m(s))$ of the non-compliant flow such that $s + t(s)$ is feasible for instance (N, r) and

$$\begin{aligned}
&\forall n \in \operatorname{Supp}(t(s)), l_n\left(s_n + t_n(s), m_n(s)\right) = l_0 \\
&\forall n \notin \operatorname{Supp}(t(s)), l_n\left(s_n, m_n(s)\right) \geq l_0
\end{aligned}$$

This extends the definition usually used in the congestion games literature, see for example [**?**]

We will denote by $\mathrm{S}(N, r, \beta) \subset \mathbb{R}^N$ the set of Stackelberg strategies for network instance (N, r, β).

7.2 Example of a feasible assignment that is not a Stackelberg strategy

Note that a feasible flow assignment s of compliant flow may fail to induce a pure Nash equilibrium (t, m).

To see this, consider the following 2-link network where such that link 1 is faster $a_1 < a_2$ and has larger capacity $x_1^{\max} > x_2^{\max}$. Now assume that the network is subject to flow demand $r = x_1^{\max} + \epsilon$ and most of the flow is compliant $\beta r = x_1^{\max}$. Consider the following assignment: $s = (x_1^{\max}, 0)$.

Assuming that the assignment of compliant agents is not affected by introducing the non-compliant flow, we have for any assignment t of non-compliant flow, $t_1 = 0$ and $t_2 > 0$. Therefore t is not at Nash equilibrium since $\operatorname{Supp}(t) = \{2\}$ and $l_2(s_2 + t_2, m_2) > l_1(s_1, 0)$ (non compliant agents are forced to use less efficient link 2).

7.3 Characterization of the induced Nash Equilibrium

We next show that the induced Nash equilibrium has one link in free-flow:

Lemma 9. *Let $s \in S(N, r, \beta)$ be a Stackelberg strategy for network instance (N, r, β), and $(t(s), m(s))$ its induced best Nash equilibrium. Then $m_{\max Supp(t(s))} = 0$.*

In other words, the last link in the support of $t(s)$ is in free-flow.

Proof. Note that $t(s)$ is an assignment at Nash for instance $(N, \beta r)$ and latencies

$$\begin{aligned}
\tilde{l}_n : [0, x_n^{\max} - s_n] \times \{0, 1\} &\longrightarrow \mathbb{R}_+ \\
(x_n, m_n) &\longmapsto l_n(s_n + x_n, m_n)
\end{aligned}$$

by Theorem1, we immediately have $m_{\max \operatorname{Supp}(t(s))} = 0$. ☐

7.4 Optimal Stackelberg strategies

In this section we solve for optimal Stackelberg strategies, i.e. Stackelberg strategies that induce Nash equilibria of minimal cost. This is described by the following optimization problem

$$\min_{s \in \mathrm{S}(N,r,\beta)} C\left(s + t(s), m(s)\right)$$

where s is a Stackelberg strategy and $(t(s), m(s))$ is the non-compliant flow assignment at the equilibrium induced by s.

We define an optimal Stackelberg strategy s^* to be a solution to the optimization problem

$$s^* = \arg \min_{s \in \mathrm{S}(N,r,\beta)} C\left(s + t(s), m(s)\right)$$

and a Stackelberg equilibrium to be the Nash equilibrium $(t(s^*), m(s^*))$ induced by optimal strategy $s*$.

8 Computing the optimal Stackelberg strategy

In this section, we show the following result: the optimal Stackelberg strategy can be computed in polynomial time for parallel networks with N links for the class of congestion functions defined in 3.3. This result contrasts with the class of non-decreasing latency functions where the optimal Stackelberg strategy is shown to be NP-hard to compute, see [**?**].

The optimal Stackelberg strategy in our case corresponds to:

- First computing the best Nash equilibrium of non-compliant users alone, $(\bar{t}, \bar{m}) = \mathrm{BNE}\left(N, (1-\beta)r\right)$
- Then assigning the compliant flow by filling the remaining links, i.e. those that are not congested under $(\bar{t}, \bar{m})$, up to maximum capacity, starting with the faster links.

Intuitively, the best induced Nash equilibrium $t(s)$ of any Stackelberg strategy s will be more congested than the best Nash equilibrium $(\bar{t}, \bar{m})$ of instance $(N, (1-\beta)r)$ (when the non-compliant flow is is the only flow on the network). So if we can find a strategy $\bar{s}$ that induces equilibrium $(\bar{t}, \bar{m})$ and that has minimal cost, then one expects this strategy to be optimal. Next, we detail this idea by defining a candidate Stackelberg strategy $\bar{s}$ that will later be shown to be optimal.

We first introduce a definition that will be useful in proving the main result. Consider a network under feasible flow assignment (x, m). A link n is said to be *exactly i-congested* if it is congested ($m_n = 1$) and its latency is exactly a_i. It is said to be *at least i-congested* ($i \geq n+1$) if n is congested and its latency is at least a_i.

Definition 5. Link n is exactly i-congested ($i \geq n+1$) under assignment (x, m) if

$$\begin{aligned} m_n &= 1 \\ l_n(x_n, m_n) &= a_i \end{aligned}$$

which is equivalent to

$$\begin{aligned} m_n &= 1 \\ x_n &= \hat{x}_n\left(i\right) \end{aligned}$$

Note that in a network under best Nash assignment (t, m), if $i = \max \mathrm{Supp}(t)$, then all links $n \in \{1, \dots, i-1\}$ are exactly i-congested.

Definition 6. Link n is at least i-congested ($i \geq n+1$) under assignment (x, m) if

$$\begin{aligned} m_n &= 1 \\ l_n(x_n, m_n) &\geq a_i \end{aligned}$$

which is equivalent to

$$\begin{aligned} m_n &= 1 \\ x_n &\leq \hat{x}_n(i) \end{aligned}$$

Note that if $j \geq i \geq n+1$, then if link n is at least j-congested under (x, m), then it is also at least i-congested under (x, m).

8.1 A candidate Stackelberg strategy: Non-Compliant First

Let $(\bar{t}, \bar{m})$ denote the best Nash equilibrium for the instance $(N, (1-\beta)r)$. Let $k = \max \operatorname{Supp}(\bar{t})$. We have

$$\bar{m} = (1, \ldots, 1, \overset{k}{0}, \ldots, 0)$$

and

$$\bar{t} = \left(\hat{x}_1(k), \ldots, \hat{x}_{k-1}(k), (1-\beta)r - \sum_{n=1}^{k-1} \hat{x}_n(k), 0, \ldots, 0\right)$$

i.e. links $\{1, \ldots, k-1\}$ are k-congested, and link k is in free-flow. Figure 6 shows the assignment at Nash equilibrium $(\bar{t}, \bar{m})$ on a sample network, where the latency in congestion $l_n(x_n, 1)$ is taken to be affine for simplicity.

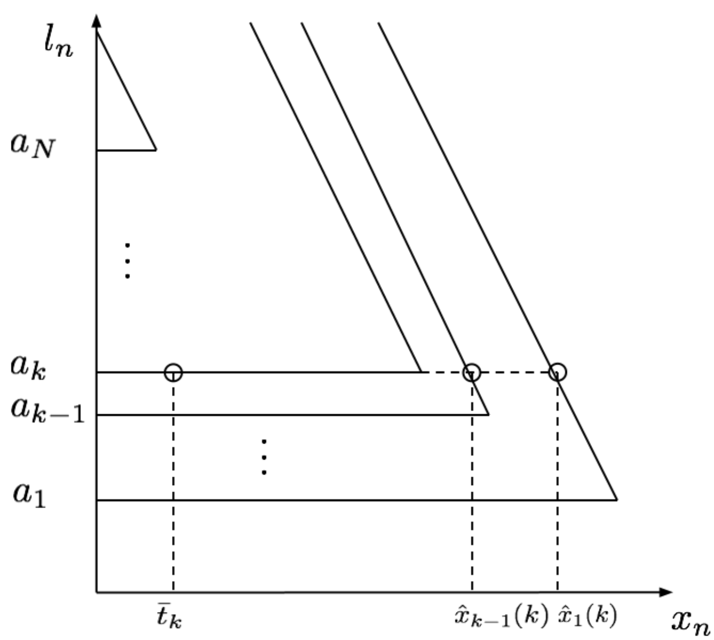


Figure 6: Best Nash equilibrium $(\bar{t}, \bar{m})$ of non-compliant flow $(1-\beta)r$. All links in the support $\{1, \ldots, k\}$ have the same latency a_k. Links $\{1, \ldots k-1\}$ are exactly k-congested, and link k is in free-flow.

Now define Stackelberg strategy $\bar{s}$ as the optimal assignment of compliant flow βr that induces equilibrium $(\bar{t}, \bar{m})$. It is easy to see that $\bar{s}$ is simply given by assigning the compliant flow to remaining links $\{k, k+1, \dots, N\}$ successively, each up to maximum capacity. The strategy $\bar{s}$ will assign $x_k^{\max} - t_k$ on link k, then $x_{k+1}^{\max}$ on link $k+1$, $x_{k+2}^{\max}$ on link $k+2$ and so on. Let $l = \min\{n | \beta r - (\sum_{n=k}^{l-1} x_n^{\max} - t_k) \geq 0\}$ be the least efficient link used by the Stackelberg assignment. Then $\bar{s}$ is given by

$$\bar{s} = \left(0, \dots, \overset{k-1}{0}, x_k^{\max} - \bar{t}_k, x_{k+1}^{\max}, \dots, x_{l-1}^{\max}, \beta r - (\sum_{n=k}^{l-1} x_n^{\max} - t_k), \overset{l+1}{0}, \dots, 0\right) \tag{10}$$

Equivalently, the total assignment $\bar{x} = \bar{s} + \bar{t}$ is given by

$$\bar{x} = \left(\hat{x}_1(k), \dots, \hat{x}_{k-1}(k), x_k^{\max}, x_{k+1}^{\max}, \dots, x_{l-1}^{\max}, r - \sum_{n=1}^{k-1} \hat{x}_n(k) - \sum_{n=k}^{l-1} x_n^{\max}, \overset{l+1}{0}, \dots, 0\right) \tag{11}$$

and the corresponding latencies are

$$\bar{l} = \left(a_k, \dots, a_k, a_{k+1}, \dots, a_l, \overset{l+1}{0}, \dots, 0\right) \tag{12}$$

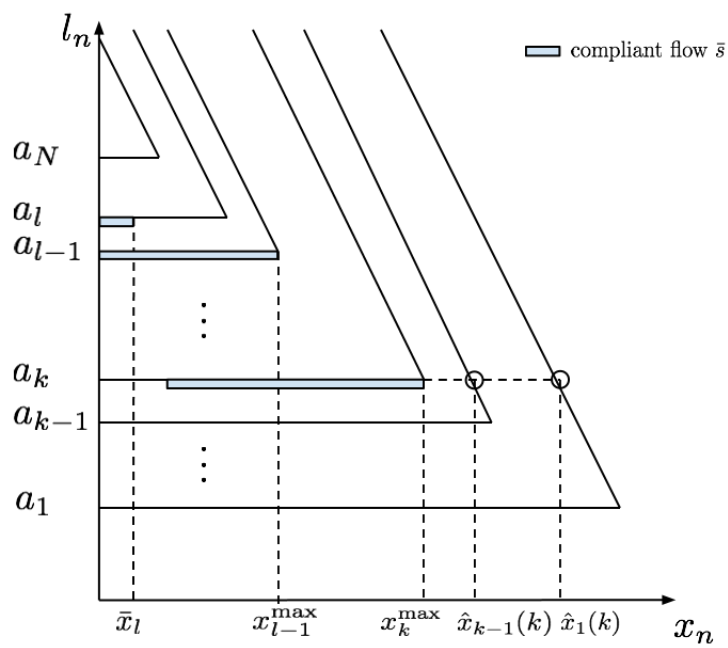


Figure 7: Equilibrium induced by candidate Stackelberg strategy $\bar{s}$. Flows that correspond to the Stackelberg strategy $\bar{s}$ are highlighted.

Figure 7 shows the total flow $\bar{x}_n = \bar{s}_n + \bar{t}_n$ on each link. Links $\{1, \dots, k-1\}$ are exactly k-congested, links $\{k, \dots, l-1\}$ are in free-flow and at maximum capacity, and the remaining flow goes on link l.

In the next section we show that strategy $\bar{s}$ is indeed an optimal Stackelberg strategy.

8.2 The Non-Compliant First strategy is optimal

Theorem 2. *$\bar{s}$ is an optimal Stackelberg strategy.*

Proof. Let $s \in S(N, r, \beta)$ be a Stackelberg strategy for instance (N, r, β) and (t, m) be the best induced Nash equilibrium for the non-compliant flow. We will show that $C(x, m) \geq C(\bar{x}, \bar{m})$, where $x = s + t$ and $\bar{x} = \bar{s} + \bar{t}$.
The proof proceeds as follows: we first show that links $\{1, \dots, l-1\}$ are more congested under assignment (x, m) than under $(\bar{x}, \bar{m})$, in the following sense: these links have worse latency

$$l_n(x_n, m_n) \geq l_n(\bar{x}_n, \bar{m}_n) \qquad \forall n \in \{1, \dots, l-1\}$$

and hold less flow

$$x_n \leq \bar{x}_n \qquad \forall n \in \{1, \dots, l-1\}$$

Then we conclude by lower bounding the cost $C(x, m)$.

Let $k' = \max \mathrm{Supp}(t)$ be the link with largest free-flow latency, in the support of the non-compliant flow. By Lemma 9, we have $m_{k'} = 0$, i.e. link k' is in free-flow under assignment (x, m). We start by showing that $k' \geq k$ where $k = \max \mathrm{Supp}(\bar{t})$

Lemma 10. *The last link in the support of t is less efficient than the last link in the support of $\bar{t}$. $k' \geq k$*

Intuitively, since $\bar{t}$ is the best Nash equilibrium of the non-compliant agents when they are not sharing the network with any other flow, the assignment $(\bar{t}, \bar{m})$ will use more efficient links than any Nash assignment (t, m) of non-compliant agents when they are sharing the network with additional flow s.

Proof. First note that $s + t$ restricted to $\mathrm{Supp}(t)$ is at Nash equilibrium, and it is by definition the best such Nash equilibrium. Then since link k' is in free-flow and $k' \in \mathrm{Supp}(t)$, the cost of this Nash equilibrium is simply $r' a_{k'}$ where $r' = \sum_{n \in \mathrm{Supp}(t)} s_n + t_n$ is the total flow restricted to $\mathrm{Supp}(t)$.

We have $\bar{t}$ is the best Nash equilibrium of flow $(1-\beta)r$ with cost $(1-\beta) r a_k$, and $(1-\beta)r = \sum_{n \in \mathrm{Supp}(t)} t_n \leq \sum_{n \in \mathrm{Supp}(t)} s_n + t_n = r'$. Then since the average cost of the best Nash equilibrium is increasing in the total flow, we have

$$\frac{r' a_{k'}}{r'} \geq \frac{(1-\beta) r a_k}{(1-\beta) r}$$

Therefore we must have $a_{k'} \geq a_k$, i.e. $k' \geq k$. ☐

Using the lemma, we can now show that links $\{1, \dots, l-1\}$ are more congested under assignment (x, m) than candidate assignment $(\bar{x}, \bar{m})$.

We have from Definition 4 of a Stackelberg strategy and its induced equilibrium, $\forall n \in \{1, \dots, k'-1\}$, $l_n(x_n, m_n) \geq l_{k'}(x_{k'}, m_{k'}) \geq a_{k'}$, i.e. $\forall n \in \{1, \dots, k'-1\}$, n is at least k'-congested under assignment (x, m). We also have by (12) $\forall n \in \{1, \dots, k-1\}$, n is exactly k-congested under assignment $(\bar{x}, \bar{m})$. Thus using the fact that $k' \geq k$

$$l_n(x_n, m_n) \geq a_{k'} \geq a_k = l_n(\bar{x}_n, \bar{m}_n) \qquad \forall n \in \{1, \dots, k-1\}$$
$$x_n \leq \hat{x}_n(k') \leq \hat{x}_n(k) = \bar{x}_n \qquad \forall n \in \{1, \dots, k-1\}$$

and we have from (11) that $\forall n \in \{k, \dots, l-1\}$, n is in free flow and at maximum capacity under assignment $(\bar{x}, \bar{m})$ (i.e. $\bar{x}_n = x_n^{\max}$ and $l_n(\bar{x}_n) = a_n$). Thus

$$l_n(x_n, m_n) \geq a_n = l_n(\bar{x}_n, \bar{m}_n) \qquad \forall n \in \{k, \dots, l-1\}$$
$$x_n \leq x_n^{\max} = \bar{x}_n \qquad \forall n \in \{k, \dots, l-1\}$$

Therefore we have

$$l_n(x_n, m_n) \geq l_n(\bar{x}_n, \bar{m}_n) \qquad \forall n \in \{1, \dots, l-1\} \tag{13}$$

$$x_n \leq \bar{x}_n \qquad \forall n \in \{1, \dots, l-1\} \tag{14}$$

Note that $\forall n \in \{1, \dots, k\}$, $l_n(\bar{x}_n, \bar{m}_n) = a_k \leq a_l$, and $\forall n \in \{k, \dots, l-1\}$, $l_n(\bar{x}_n, \bar{m}_n) = a_n \leq a_l$, thus we have

$$l_n(\bar{x}_n, \bar{m}_n) \leq a_l \qquad \forall n \in \{1, \dots, l-1\} \tag{15}$$

Also note that links $n \in \{l, \dots, N\}$ have latency at least a_n (the latency on a link is always greater than the free-flow latency) and $a_n \geq a_l$, thus

$$l_n(x_n, m_n) \geq a_l \qquad \forall n \in \{l, \dots, N\} \tag{16}$$

We can now lower-bound the cost of the assignment (x, m) where $x = s + t$ and (t, m) is the best Nash equilibrium induced by s. For conciseness, we use l_n as a shorthand for $l_n(x_n, m_n)$ and $\bar{l}_n$ as a shorthand for $l_n(\bar{x}_n, \bar{m}_n)$. We have

$$\begin{aligned}
C(x,m) - C(\bar{x}, \bar{m}) &= \sum_{n=1}^{N} x_n l_n - \sum_{n=1}^{l} \bar{x}_n \bar{l}_n && \text{since } \mathrm{Supp}(\bar{x}) = \{1, \dots, l\} \\
&= \sum_{n=1}^{l-1} (x_n l_n - \bar{x}_n \bar{l}_n) + \sum_{n=l}^{N} x_n l_n - \bar{x}_l a_l && \text{since } \bar{l}_l = a_l \\
&\geq \sum_{n=1}^{l-1} (x_n l_n - \bar{x}_n \bar{l}_n) + \sum_{n=l}^{N} x_n a_l - \bar{x}_l a_l && \text{using (16)} \\
&= \sum_{n=1}^{l-1} (x_n - \bar{x}_n) l_n + \sum_{n=1}^{l-1} x_n (l_n - \bar{l}_n) + \sum_{n=l}^{N} x_n a_l - \bar{x}_l a_l \\
&\geq \sum_{n=1}^{l-1} (x_n - \bar{x}_n) l_n + \sum_{n=l}^{N} x_n a_l - \bar{x}_l a_l && \text{using (13)}
\end{aligned}$$

Then using (14) and (15) we have $\forall n \in \{k, \dots, l-1\}$, $x_n - \bar{x}_n \geq 0$ and $l_n \leq a_l$, thus $\sum_{n=1}^{l-1} (x_n - \bar{x}_n) l_n \geq \sum_{n=1}^{l-1} (x_n - \bar{x}_n) a_l$. Therefore

$$\begin{aligned}
C(x,m) - C(\bar{x}, \bar{m}) &\geq \sum_{n=1}^{l-1} (x_n - \bar{x}_n) a_l + \sum_{n=l}^{N} x_n a_l - \bar{x}_l a_l \\
&= \left(\sum_{n=1}^{N} x_n - \sum_{n=1}^{l} \bar{x}_n \right) a_l \\
&= 0
\end{aligned}$$

$\square$